

Laporan Tugas Besar III IF2211 Strategi Algoritma

Semester II Tahun 2020/2021

Penerapan String Matching dan Regular Expression dalam Pembangunan Deadline Reminder Assistant

Oleh

M. Hilal Alhamdy (13519024)

Mohammad Yahya Ibrahim (13519091)

Habibina Arif Muzayyan (13519125)



PROGRAM STUDI TEKNIK INFORMATIKA

SEKOLAH TEKNIK ELEKTRO DAN INFORMATIKA

INSTITUT TEKNOLOGI BANDUNG

BANDUNG

2021

BAB 1

Deskripsi Tugas

Dalam tugas besar ini, Anda akan diminta untuk membangun sebuah chatbot sederhana yang berfungsi untuk membantu mengingat berbagai deadline, tanggal penting, dan task-task tertentu kepada user yang menggunakannya. Dengan memanfaatkan algoritma String Matching dan Regular Expression, Anda dapat membangun sebuah chatbot interaktif sederhana layaknya Google Assistant yang akan menjawab segala pertanyaan Anda terkait informasi deadline tugas-tugas yang ada.

Terdapat fitur-fitur yang digunakan dalam chatbot tersebut, Deadline Reminder Assistant. akan dibangun dengan sistem Question and Answer dimana pengembang diharapkan sudah menyediakan kumpulan formula tertentu untuk melakukan pendekripsi setiap perbedaan command atau perintah pada aplikasi Chatbot

Pada fitur pertama dibuat sebuah fungsi yang dapat menambahkan sebuah task baru berupa suatu kalimat diklasifikasi sebagai suatu task apabila mengandung tanggal, kode Mata Kuliah, jenis tugas, dan topik tugas. Ketika pengguna mengirimkan pesan dan dikenali oleh asisten dari chatbot , maka assistant chatbot akan mengirim pesan balasan berupa id, tanggal,kode mata kuliah , jenis tugas, dan topik. Pada fitur kedua dibuat sebuah fungsi yang dapat melihat daftar task yang harus dikerjakan. Chatbox dapat mengenali daftar task yang harus dikerjakan melalui inputan task yang telah dikirimkan. Pada fitur ketiga, chatbox dapat menampilkan deadline dari suatu task tertentu . ketika pengguna mengirimkan pesan pertanyaan mengenai deadline suatu tugas, chatbot akan mengirim pesan balasan berupa tanggal deadline tugas tersebut.

Pada fitur keempat dibuat sebuah fungsi yang dapat memperbarui task tertentu, fitur ini dapat memperbarui tanggal dari suatu task. Perintah yang dimasukkan meliputi 1 keyword untuk memperbaharui suatu task dan nomor task tertentu. Fitur kelima dari chatbot ini, dapat menandai bahwa suatu task sudah selesai dikerjakan. Apabila user sudah menyelesaikan suatu task, maka task tersebut bisa ditandai bahwa task tersebut sudah selesai dan tidak perlu lagi ditampilkan pada Daftar Task selanjutnya. Pada fitur keenam dari chatbot ini ,user dapat memilih opsi help yang difasilitasi oleh chatbot assistant, yang berisi command-command yang digunakan oleh user. Kemudian , apabila user salah mengirim pesan yang tidak sesuai dengan chatbot assistant,

maka akan di tampilkan pesan kesalahan. Pada fitur lain, terdapat bonus yang dimana apabila user salah memasukkan pesan, maka chatbot akan membenarkan kesalahan kata (typo) pada perintah yang ditulis pengguna.

BAB 2

Landasan Teori

2.1. Knuth-Morris-Pratt (KMP)

Algoritma Knuth-Morris-Pratt(KMP) adalah salah satu algoritma pencarian string, dikembangkan secara terpisah oleh Donald E. Knuth pada tahun 1967 dan James H. Morris bersama Vaughan R. Pratt pada tahun 1966, namun keduanya mempublikasikannya secara bersamaan pada tahun 1977 (Manikandan dan Ramyachitra,2018). Jika kita melihat algoritma KMP lebih mendalam, kita mengetahui bahwa dengan mengingat beberapa perbandingan yang dilakukan sebelumnya kita dapat meningkatkan besar pergeseran yang dilakukan. Hal ini akan menghemat perbandingan, yang selanjutnya akan meningkatkan kecepatan pencarian

2.2. Boyer-Moore

Algoritma Boyer-Moore adalah algoritma pencarian string, algoritma ini dianggap sebagai algoritma yang paling efisien pada aplikasi umum. Ide utama dari Algoritma ini adalah dengan melakukan pencocokan dari paling kanan suatu string yang dicari. Algoritma ini dapat mencari dengan lebih cepat dibandingkan dengan proses pencarian lainnya.

2.3. Regular Expression

Regular Expression atau Regex merupakan serangkaian karakter yang mendefinisikan sebuah pola pencarian kata dan lain-lain. Pola tersebut digunakan dalam string matching untuk melakukan operasi cari dan mengganti kata pada string. Ekspresi reguler merupakan teknik yang dikembangkan dalam bidang ilmu komputer teori dan teori bahasa formal.

2.4. Aplikasi Chatbot

Istilah Chatbot dengan nama lain chatterbot merupakan sebuah layanan obrolan robot/tokoh secara virtual dengan kecerdasan buatan atau AI (Artificial Intelligence) yang menirukan percakapan manusia melalui pesan suara, obrolan teks ataupun keduanya. Singkatnya Chatbot ini merupakan percakapan virtual dimana satu pihak adalah sebuah robot chat yang bertujuan untuk sarana hiburan. Fitur chatbot ini telah digunakan di berbagai industri untuk menyampaikan informasi.

Cara kerja Chatbot itu sendiri dapat memberikan tanggapan dengan berbagai macam input. Bot yang melakukannya dengan cara memindai kata kunci dalam inputan, setelah itu di balas dengan kata kunci yang paling cocok atau bisa juga dengan pola kata yang paling mirip dengan basis data tekstual

BAB 3

Analisis Pemecahan Masalah

3.1. Langkah Penyelesaian Setiap Fitur

3.1.1. Menambahkan *task* baru

- Mengambil data dari web berupa masukan dari user.
- Untuk menentukan apakah suatu masukan termasuk dalam fungsi menambahkan task baru menggunakan 2 metode. Pertama memeriksa apakah masukan mengandung komponen berupa tanggal, kode mata kuliah, jenis tugas, dan topik tugas. Selain itu dapat menggunakan metode string matching (Boyer-Moore/KMP) untuk mencari kata kunci berupa “Tambah”.
- Mendapatkan data/informasi dari 4 komponen tersebut dengan menggunakan *regular expression* dan algoritma *string matching*. Komponen seperti tanggal, kode mata kuliah, Topik tugas dicari menggunakan regular expression sedangkan komponen jenis tugas dicari menggunakan algoritma *string matching*.
- Mengecek apakah task yang ingin dimasukan sudah ada datanya di database atau belum. Pengecekan eksistensi dari sebuah data task dilakukan dengan iterasi setiap data task dalam database lalu membandingkan empat komponen
- Apabila task yang ingin ditambah belum ada di database, maka program akan menyimpan data task yang baru dan melakukan update ke dalam database. Selain itu akan mengirimkan keluaran pesan yang mengatakan bahwa task berhasil ditambahkan.
- Apabila task yang ingin ditambah sudah ada dalam database maka program akan mengirimkan pesan yang mengatakan bahwa task yang sedang ingin coba dimasukan sudah berada dalam database.

3.1.2. Melihat daftar *task* yang harus dikerjakan

- Mengambil data dari web berupa masukan dari user.
- Untuk menentukan apakah suatu masukan termasuk dalam fungsi melihat daftar *task* yang harus dikerjakan adalah dengan mencari keyword dengan string matching untuk keyword “Apa saja”.
- Dalam fungsi melihat daftar *task* yang harus dikerjakan terdapat beberapa jenis perintah yang dapat dilakukan. Seperti melihat seluruh daftar task yang sudah tercatat oleh assistant, melihat seluruh daftar task berdasarkan periode waktu (antara DATE 1 sampai DATE 2, N minggu ke depan, N hari ke depan, Hari ini), serta memfilter berdasarkan jenis task yang ingin ditampilkan.
- Untuk menentukan jenis perintah dalam fungsi melihat daftar task yang harus dikerjakan, dilakukan string matching untuk keyword berupa “antara”, “sampai”, “minggu ke depan”, “hari ke depan”, “hari ini”, ‘sejauh ini’.

- Setelah mendapatkan jenis perintah yang sesuai dengan masukan, mendapatkan data/informasi komponen berupa tanggal periode yang ingin diketahui deadline-nya
- Iterasi dari setiap data task dalam database untuk mencari data yang memenuhi syarat dari data periode waktu yang telah didapatkan (Tanggal). Selain itu untuk masukan yang mempunyai tambahan atribut berupa jenis task akan diproses juga dalam proses iterasi
- Apabila ditemukan data yang memenuhi syarat maka program akan mengembalikan hasil kumpulan task yang memenuhi syarat tersebut.
- Apabila pada proses iterasi tidak ditemukan data task yang memenuhi maka program akan mengembalikan pesan yang memberikan pesan bahwa tidak ada task yang harus dikerjakan

3.1.3. Menampilkan *deadline* dari suatu *task* tertentu

- Mengambil data dari web berupa masukan dari user.
- Untuk menentukan apakah suatu masukan termasuk dalam fungsi Menampilkan *deadline* dari suatu *task* tertentu adalah dengan mencari keyword dengan string matching untuk keyword “Kapan”.
- Mendapatkan data/informasi komponen berupa kode mata kuliah yang ingin diketahui
- Iterasi dari setiap data task dalam database untuk mencari data kode mata kuliah hasil yang diinginkan dan mengecek apakah *task* tersebut sudah selesai dikerjakan atau belum. Task yang masuk dalam kategori maka akan dimasukan ke dalam sebuah list *result*
- Apabila tidak ditemukan deadline dari suatu task yang diinginkan maka program akan mengembalikan pesan bahwa tidak ada task task sedangkan apabila ada task deadline task kode kuliah yang diinginkan dalam database maka akan memberikan pesan berupa deadline task hasil iterasi tersebut.

3.1.4. Memperbarui task tertentu

- Mengambil data dari web berupa masukan dari user.
- Untuk menentukan apakah suatu masukan termasuk dalam fungsi memperbarui deadline *task* tertentu adalah dengan mencari keyword dengan string matching untuk keyword “Diundur”.
- Mendapatkan data/informasi komponen berupa tanggal dan ID dari task yang ingin diganti deadline-nya. Proses ini dilakukan dengan menggunakan regular expression.
- Iterasi untuk setiap ID task dalam database. Apabila ditemukan ID task yang ingin diganti deadline-nya maka program akan mengembalikan pesan yang menyatakan bahwa memperbarui deadline suatu task telah berhasil dilakukan.

- Apabila tidak ditemukan ID task yang ingin diganti deadline-nya, maka program akan mengembalikan pesan yang memberitahu bahwa ID task yang dimaksud tidak dapat ditemukan dalam database.
- 3.1.5. Menandai bahwa suatu task sudah selesai dikerjakan
 - Mengambil data dari web berupa masukan dari user.
 - Untuk menentukan apakah suatu masukan termasuk dalam fungsi menandai bahwa suatu task sudah selesai dikerjakan adalah dengan mencari keyword dengan string matching untuk keyword “Selesai”.
 - Mendapatkan data/informasi komponen berupa tanggal dan ID dari task yang ingin ditandai sudah selesai dikerjakan. Proses ini dilakukan dengan menggunakan regular expression.
 - Iterasi untuk setiap ID task dalam database. Apabila ditemukan ID task yang ingin ditandai sudah selesai dikerjakan maka program akan mengembalikan pesan yang menyatakan bahwa proses menandai suatu task sudah selesai dikerjakan telah berhasil dilakukan.
 - Apabila tidak ditemukan ID task yang ingin ditandai sudah selesai dikerjakan, maka program akan mengembalikan pesan yang memberitahu bahwa ID task yang dimaksud tidak dapat ditemukan dalam database.
- 3.1.6. Menampilkan opsi help yang difasilitasi oleh assistant
 - Mengambil data dari web berupa masukan dari user.
 - Untuk menentukan apakah suatu masukan termasuk dalam fungsi menampilkan opsi help yang difasilitasi oleh assistant adalah dengan mencari keyword dengan string matching untuk keyword “Help” atau“ Apa yang bisa Assistant lakukan”.
 - Apabila keyword tersebut ditemukan, maka program akan mengembalikan pesan yang berisi berupa daftar fitur serta daftar kata penting yang dimiliki oleh program
- 3.1.7. Mendefinisikan list kata penting terkait apakah itu merupakan suatu task atau tidak dan menampilkan pesan error jika assistant tidak dapat mengenali masukan user.
 - Mengambil data dari web berupa masukan dari user.
 - Untuk menentukan apakah suatu masukan termasuk dalam fungsionalitas program atau tidak adalah dengan iterasi setiap keyword yang tersedia dengan string matching.
 - Apabila tidak ada keyword yang cocok maka program akan mengembalikan pesan yang mengatakan bahwa masukan yang diterima tidak dapat dikenali oleh program

3.2. Fitur fungsional dan arsitektur Chatbot yang dibangun

- 3.2.1. Menambahkan *task* baru

Fitur Menambahkan task baru merupakan fitur yang berfungsi untuk menambahkan task baru dalam database program. Supaya dapat bekerja, fungsi ini mempunyai beberapa kebutuhan (requirement). Kebutuhan yang wajib yaitu masukan harus memiliki empat komponen berupa tanggal, kode mata kuliah, jenis tugas, dan topik tugas. Kebutuhan yang tidak wajib adalah penggunaan keyword “Tambah” dalam masukan yang diberikan ke program.

3.2.2. Melihat daftar *task* yang harus dikerjakan

Fitur Melihat daftar task yang harus dikerjakan merupakan fitur yang berfungsi untuk menampilkan deadline-deadline dari task. Deadline task-task yang ingin ditampilkan dapat dikustomisasi dengan berbagai parameter. Seperti menampilkan deadline berdasarkan periode waktu (Jangka 2 waktu, N minggu ke depan, N hari ke depan) dan berdasarkan tipe task.

3.2.3. Menampilkan *deadline* dari suatu *task* tertentu

Fitur menampilkan deadline dari suatu task tertentu merupakan fitur yang berfungsi untuk menampilkan deadline dari task X. Task X diberitahu secara eksplisit dalam masukan dari user.

3.2.4. Memperbarui task tertentu

Fitur memperbarui task tertentu adalah suatu fitur yang berfungsi untuk memperbarui deadline suatu task dengan ID X. Dengan X merupakan ID dari suatu task yang terdaftar dalam database.

3.2.5. Menandai bahwa suatu task sudah selesai dikerjakan

Fitur menandai bahwa suatu tasks sudah selesai dikerjakan adalah suatu fitur yang berfungsi untuk memberi tanda bahwa suatu task dengan ID X telah selesai dikerjakan. Dengan X merupakan ID dari suatu task yang terdaftar dalam database

3.2.6. Menampilkan opsi help yang difasilitasi oleh assistant

Fitur menampilkan opsi help yang difasilitasi oleh assistant adalah suatu fitur yang berfungsi untuk menampilkan user fitur-fitur yang dimiliki program dan kata-kata penting yang dimiliki oleh program.

3.2.7. Mendefinisikan list kata penting terkait apakah itu merupakan suatu task

Mendefinisikan list kata penting dilakukan secara hardcoded dan list kata penting dapat dilihat oleh user dengan menggunakan fitur opsi help yang disediakan oleh program.

3.2.8. Menampilkan pesan error jika assistant tidak dapat mengenali masukan user.

Fitur menampilkan pesan error merupakan fitur yang berfungsi untuk memberikan pengguna pesan error ketika masukan yang diberikan oleh pengguna tidak dapat/gagal diproses oleh program. Hal ini terjadi karena tidak adanya keyword atau list kata penting dalam masukan pengguna.

BAB 4

Implementasi dan pengujian

4.1. Spesifikasi teknis program

Nama	Jenis	Keterangan
getRawString(String)	Fungsi	Fungsi yang menerima parameter string dan mengembalikan string tersebut dalam bentuk raw string
findLastOccurrence((String, String)	Fungsi	Fungsi yang menerima string dan char yang mengembalikan index kemunculan terakhir suatu char dalam suatu string
bmMatch (String, String)	Fungsi	Fungsi yang menerima 2 buah string yang mengembalikan index kemunculan string text dalam string pattern
findQueueMonth (String)	Fungsi	Menerima string berupa nama bulan yang selanjutnya mengembalikan string berupa urutan bulan masehi
textToAngka (String)	Fungsi	Menerima string berupa ejaan angka (Satu, dua, tiga, dll) yang selanjutnya mengembalikan dalam bentuk angka (1,2,3)
removeSpaceStartEnd (String)	Fungsi	Mengeliminasi whitespace pada awal dan akhir sebuah string
readFileTXT (String)	Fungsi	Fungsi untuk membaca dari sebuah file .txt. Mengembalikan dalam bentuk list
writeListToTXT (List,	Prosedur	Prosedur untuk membaca

String)		sebuah list ke sebuah file .txt
writeDatabase (String)	Prosedur	Prosedur untuk yang berfungsi untuk menulis list database ke file database external
readDatabase (String)	Prosedur	Prosedur untuk yang berfungsi untuk membaca file database external .txt
normalizeDatePattern (String)	Fungsi	Fungsi yang menerima parameter berupa tanggal(date) dalam berbagai bentuk aturan dan mengembalikan dalam bentuk MM/DD/YYYY
isNumber (String)	Fungsi	Mengembalikan boolean yang mengecek apakah parameter berupa string dengan jenis digital[0-9] atau tidak
findPeriodeWaktu (String)	Fungsi	Fungsi yang mengembalikan jenis periode waktu yang terkandung dalam suatu parameter (“Antara”, “N Minggu ke depan”, “N hari ke depan”, “hari ini”)
findDate (String)	Fungsi	Fungsi yang menerima parameter berupa string dan mengembalikan list yang berisi kumpulan tanggal yang terkandung dalam parameter string masukan tersebut
findMatkul (String)	Fungsi	Fungsi yang menerima parameter berupa string dan mengembalikan list yang berisi kumpulan nama matkul atau kode matkul yang terkandung dalam

		parameter string masukan tersebut. Nama/kode mata kuliah mempunyai pola XXYYYY
findTipeReminder (String)	Fungsi	Fungsi yang menerima parameter berupa string dan mengembalikan jenis task yang sesuai yang terkandung dalam parameter string tersebut (String matching)
findTopic (String)	Fungsi	Fungsi yang menerima parameter berupa string dan mengembalikan topik reminder yang berada dalam string parameter masukan tersebut
resultToString (String)	Fungsi	Menerima input result berupa list dan mengembalikan string dengan newline berupa “</p><p class=“botText“>” agar sesuai dengan frontend
addReminder (String)	Fungsi	Fungsi yang menerima masukan parameter string yang berfungsi untuk menambahkan task dan menyimpannya dalam database
showAllTask (String)	Fungsi	Fungsi yang menerima masukan parameter string dan berfungsi untuk mengembalikan daftar task yang perlu dikerjakan
periodeTask (String)	Fungsi	Fungsi yang menerima masukan parameter string dan berfungsi untuk mengembalikan daftar task yang perlu dikerjakan

		dengan adanya filter tertentu seperti (periode waktu antara 2 tanggal, N hari ke depan, N minggu ke depan, dan hari ini)
showAllTaskWithType (String)	Fungsi	Fungsi yang menerima masukan parameter string dan berfungsi untuk mengembalikan daftar task yang perlu dikerjakan dengan tambahan filter tipe task harus sesuai dengan yang ada di masukan pengguna
periodeTaskWithType (String)	Fungsi	Fungsi yang menerima masukan parameter string dan berfungsi untuk mengembalikan daftar task yang perlu dikerjakan dengan adanya filter tertentu seperti (periode waktu antara 2 tanggal, N hari ke depan, N minggu ke depan, dan hari ini) dengan tambahan filter tipe task harus sesuai dengan yang ada di masukan pengguna.
showListOfTask (String)	Fungsi	Fungsi yang menerima parameter string masukan yang selanjutnya menentukan apakah proses menampilkan daftar task memiliki parameter jenis task atau tidak
showTaskDeadlines (String)	Fungsi	Fungsi yang menerima parameter string yang berfungsi untuk mengembalikan list deadline sebuah mata kuliah yang disebutkan dalam parameter string masukan.

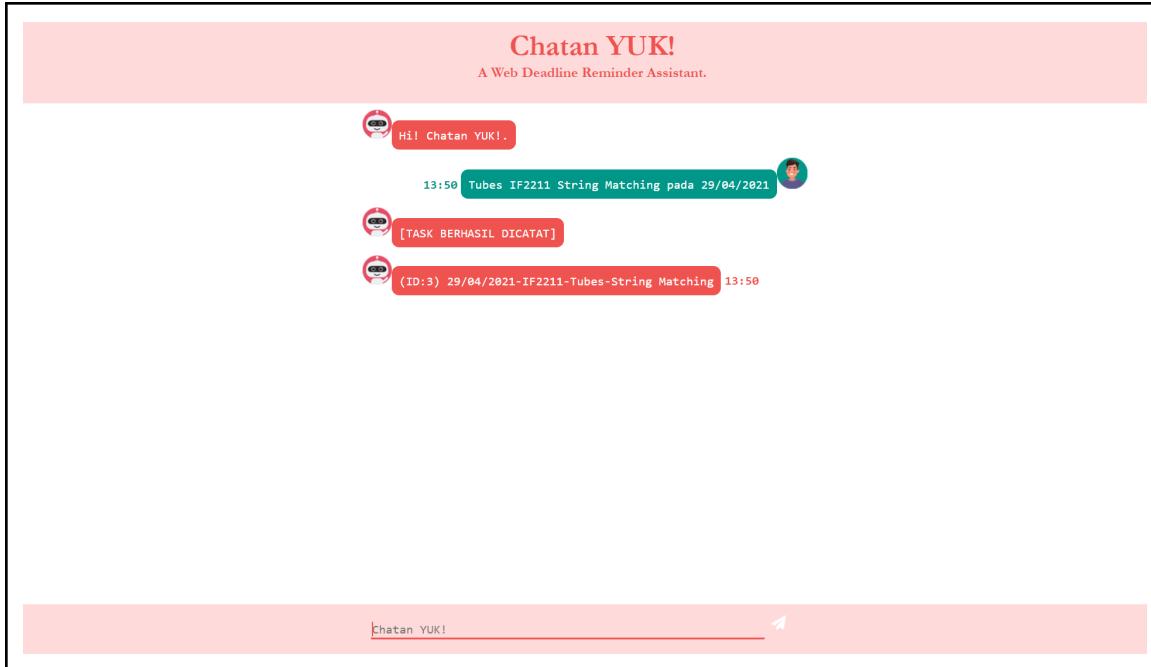
renewDeadline (String)	Fungsi	Fungsi yang menerima parameter string masukan yang selanjutnya mengubah deadline dari sebuah <i>task</i> . Mengembalikan string yang memberikan pesan bahwa suatu task sudah selesai dikerjakan dan sebaliknya apabila gagal.
markTask (String)	Fungsi	Fungsi yang menerima parameter string masukan yang selanjutnya mengubah status dari sebuah <i>task</i> menjadi sudah selesai dikerjakan. Mengembalikan string yang memberikan pesan bahwa suatu task sudah selesai dikerjakan
Main (String)	Fungsi	Fungsi main program

4.2. Tata cara penggunaan program

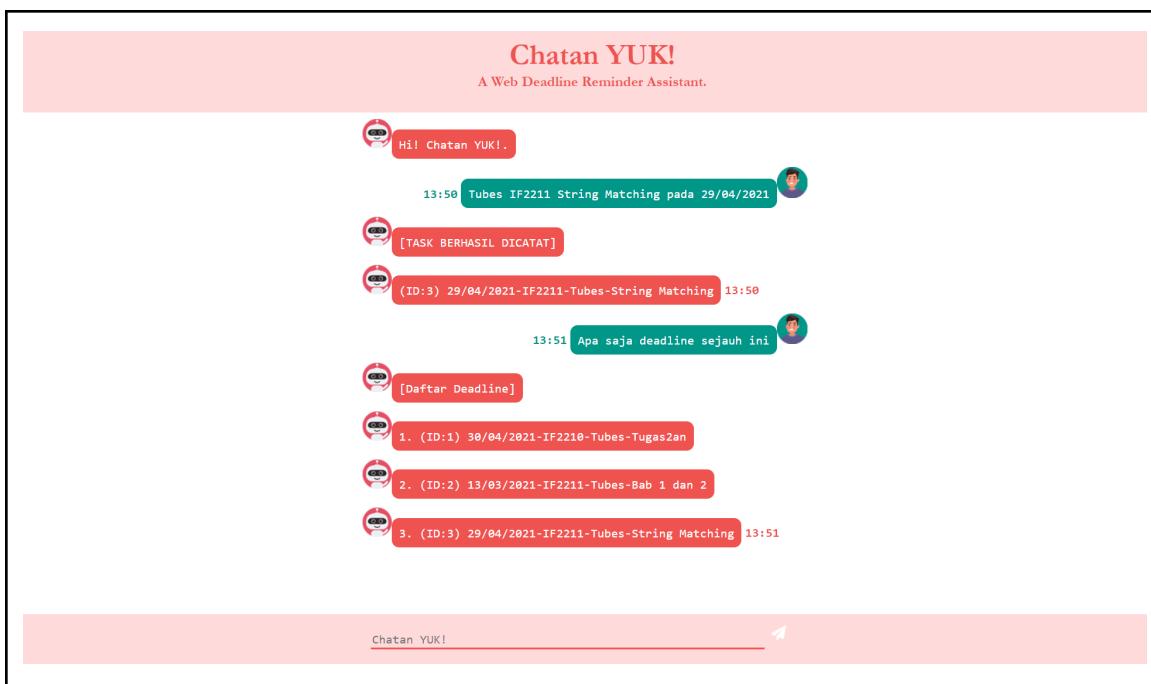
1. Jalankan program dalam lokal dengan menjalankan program app.py dan buka pranala <http://127.0.0.1:5000/> pada browser anda. Program juga telah dideploy ke website dengan membuka pranala habibinaarif.pythonanywhere.com
2. Masukkan pesan yang akan dikirimkan
3. Program akan mengirimkan balasan pesan

4.3. Hasil pengujian

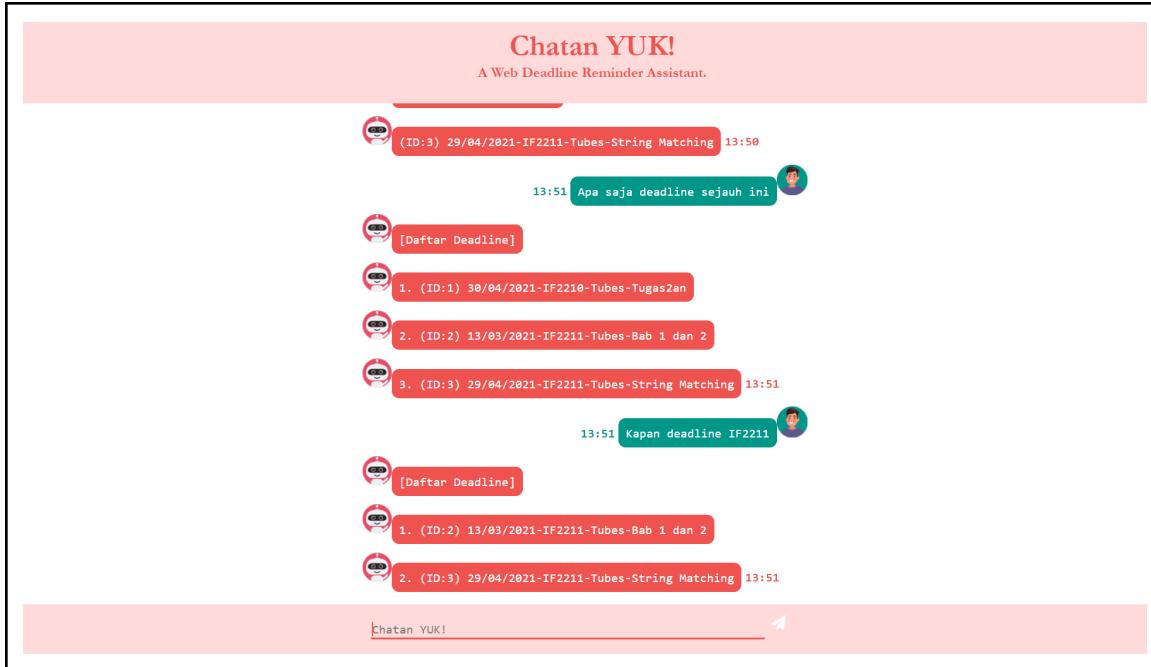
- 4.3.1. Menambahkan task baru



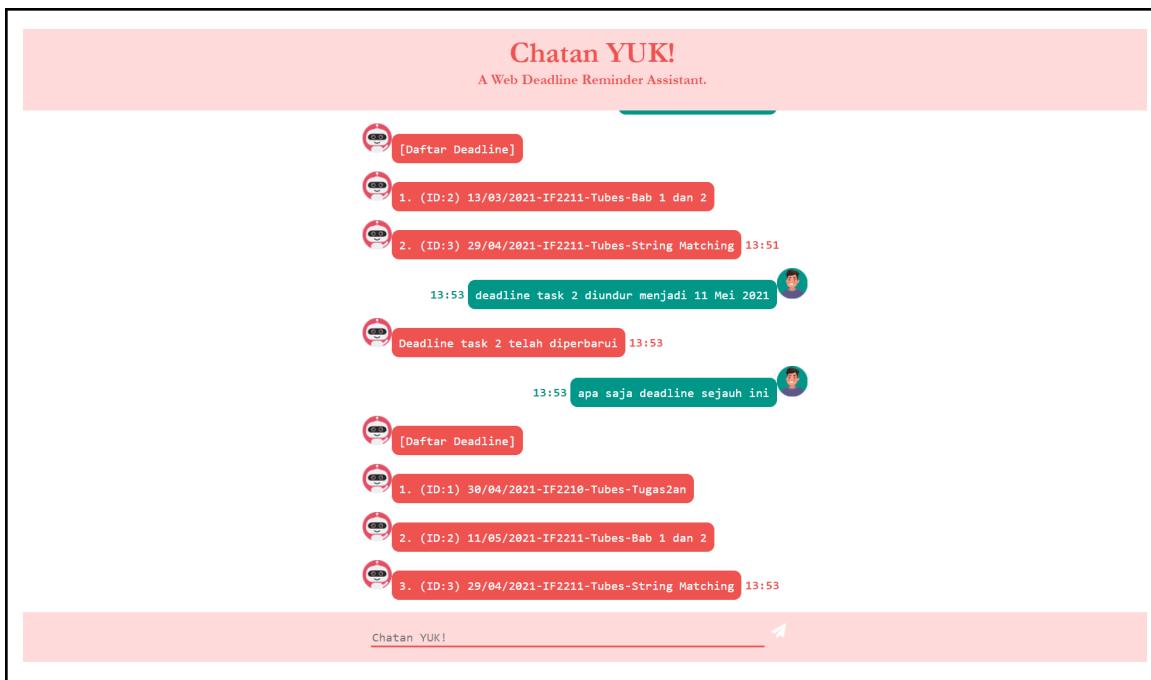
4.3.2. Melihat daftar task yang harus dikerjakan



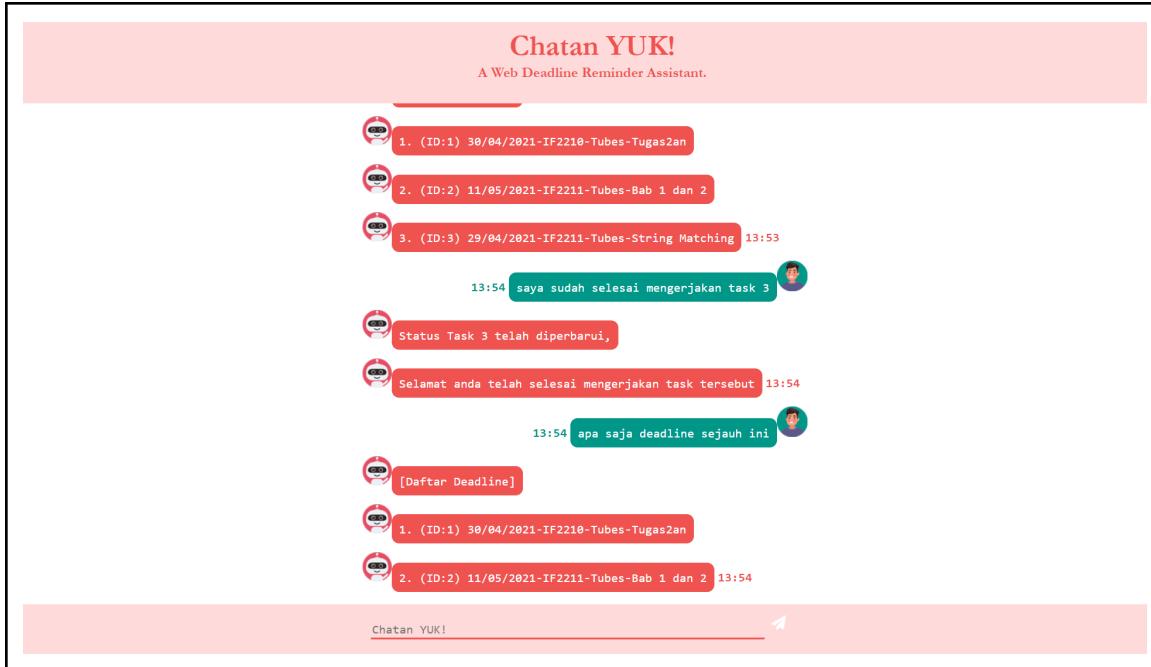
4.3.3. Menampilkan deadline dari suatu task tertentu



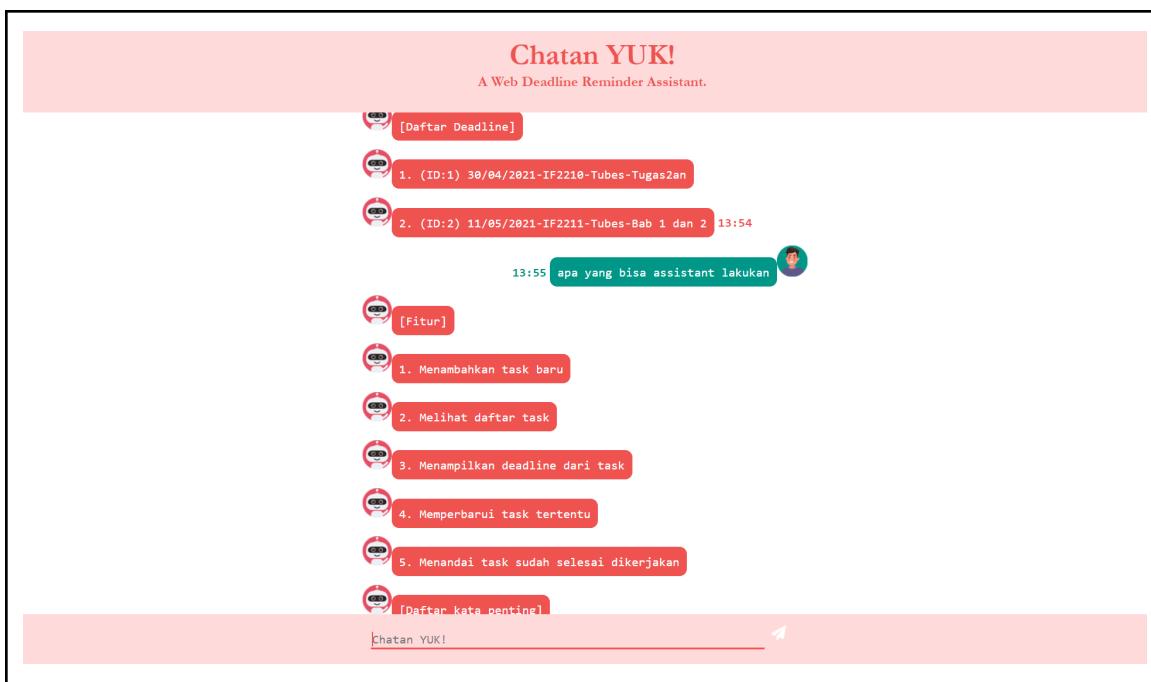
4.3.4. Memperbarui task tertentu



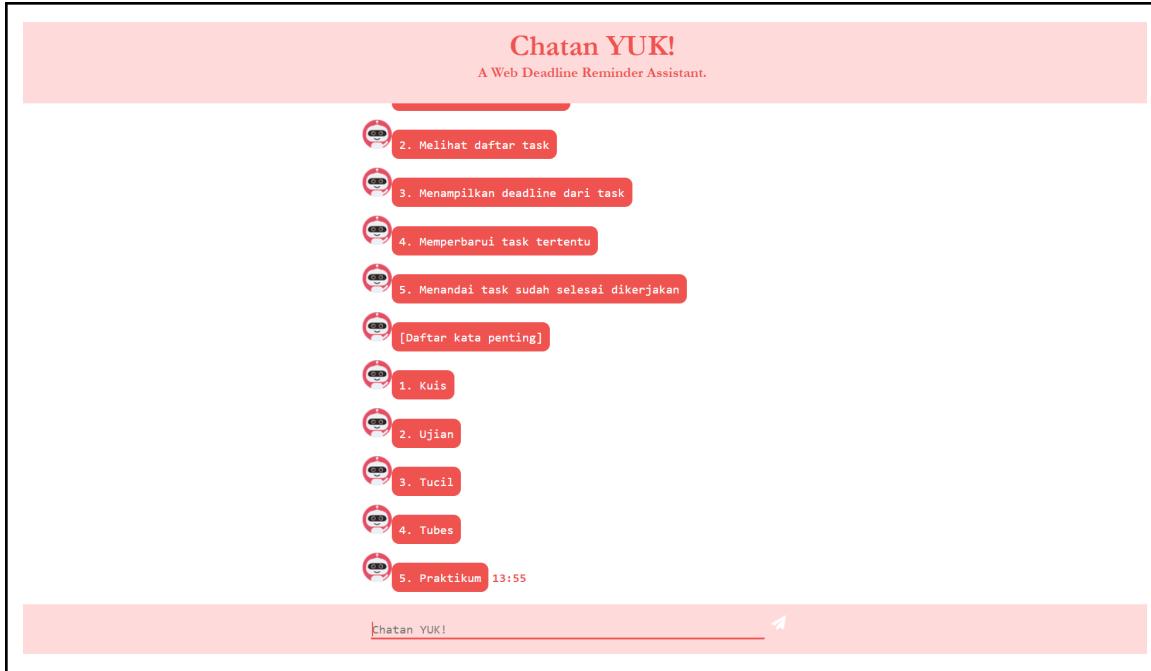
4.3.5. Menandai bahwa suatu task sudah selesai dikerjakan



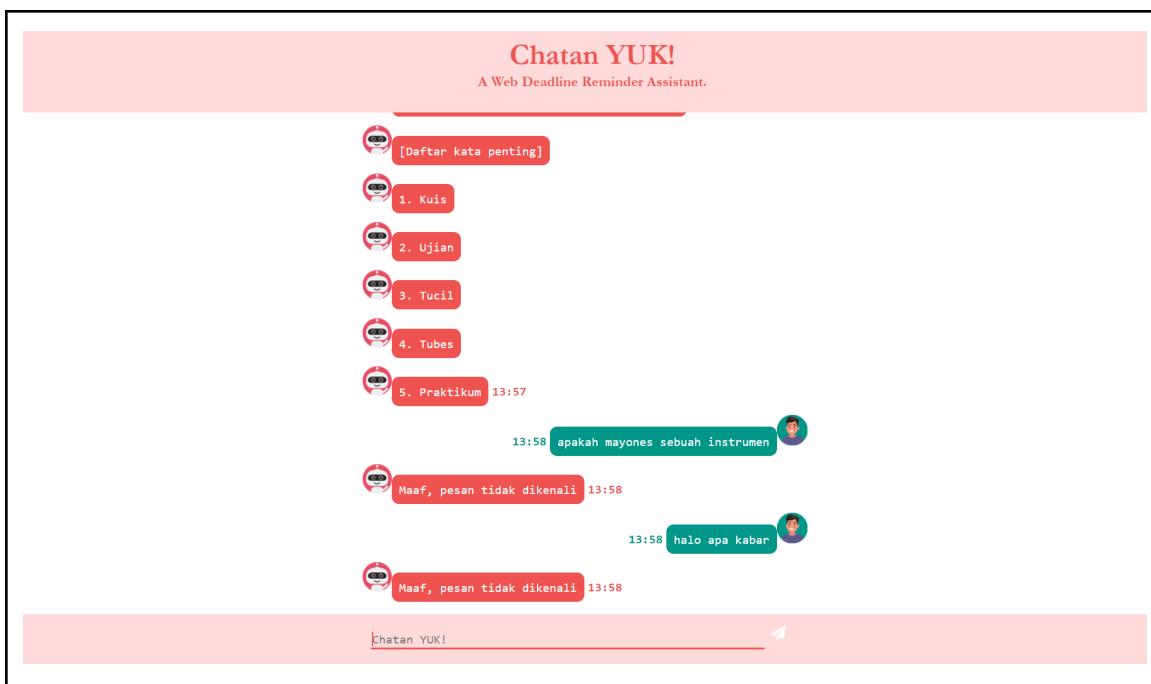
4.3.6. Menampilkan opsi help yang difasilitasi oleh assistant



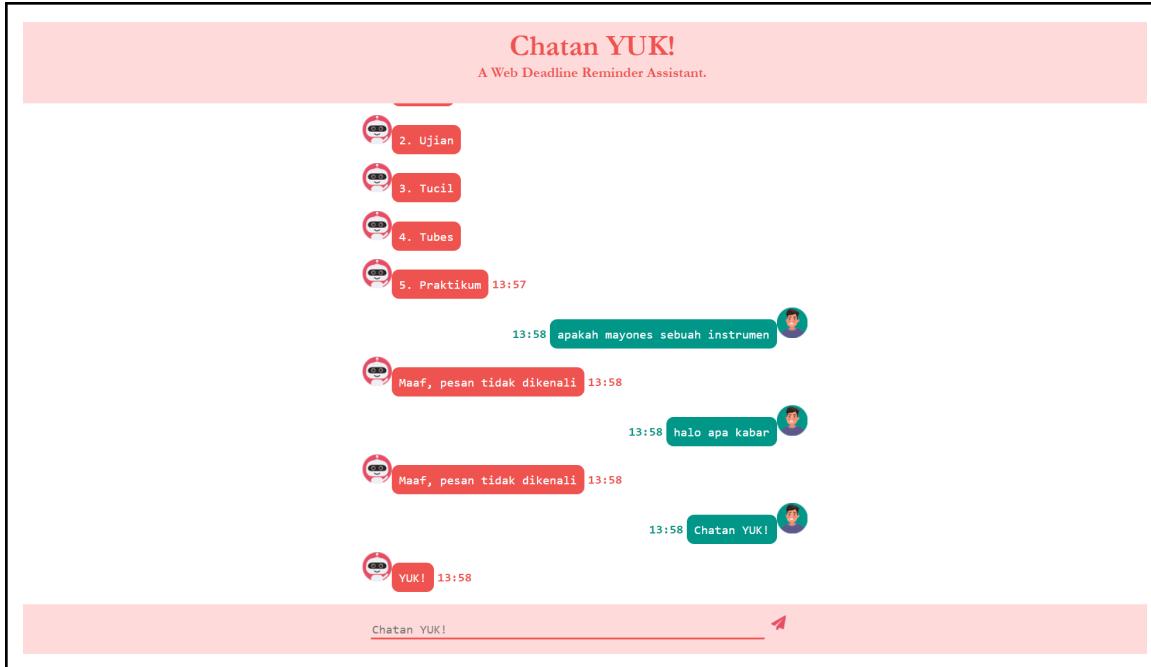
4.3.7. Mendefinisikan list kata penting terkait apakah itu merupakan suatu task atau tidak



4.3.8. Menampilkan pesan error jika assistant tidak dapat mengenali masukan user.



4.3.9. Chatan YUK!



4.4. Analisis hasil pengujian

Dari hasil percobaan yang telah dilakukan seluruh kasus yang diminta dapat berjalan dengan baik dan memberikan hasil/keluaran yang sesuai dengan harapan. Algoritma string matching membantu pada saat memproses mencari exact matching dari keyword yang dibutuhkan oleh program untuk menentukan proses apa selanjutnya yang harus diterapkan terhadap masukan pengguna. Selain itu penggunaan library regular expression juga membantu program dalam mencari data-data atau informasi yang berpola seperti tanggal dan nama mata kuliah. Selain itu regular expression juga membantu dalam menangani penggunaan huruf kapital dan kecil pada saat proses string matching. Penggunaan string matching dengan metode Boyer-Moore diharapkan akan membantu apabila skala program diperbesar sehingga proses string matching akan berjalan lebih cepat.

BAB 5

Simpulan

5.1. Kesimpulan

Program yang kami buat dapat dijalankan di terminal lokal melalui sebuah alamat link. Selain itu program yang dibuat juga sudah *di-deploy* ke sebuah hosting. Program tersebut dapat menerima sebuah masukan lalu program memproses masukan menggunakan algoritma *string matching* (KMP/Boyer-Moore). Setelah itu mengembalikan hasil sesuai dengan permintaan user.

5.2. Saran

Program yang dibuat sudah baik. Tetapi perlu ditingkatkan lagi seperti penambahan fitur yang menangani kemungkinan adanya kesalahan dalam pengetikan masukan oleh pengguna (Salah ketik). Selain itu perancang program mengharapkan modul *string matching* yang telah dibuat dapat digunakan kembali untuk tugas-tugas besar atau proyek lainnya yang membutuhkan proses *string matching*.

5.3. Refeleksi

Pada penggerjaan tugas besar ketiga ini, pada awalnya kami merasa kesulitan karena harus menangani kasus yang cukup banyak dan beragam sehingga butuh waktu untuk mendapatkan pola yang bagus agar semua kasus dapat ditangani.

Daftar Pustaka

- [1] Munir, Rinaldi. 2021. Pencocokan-String-dengan-Regex-2021.
<https://informatika.stei.itb.ac.id/~rinaldi.munir/Stmik/2018-2019/String-Matching-dengan-Regex-2019.pdf>.diakses 26 April 2021 pukul 10.22 WIB
- [2] Munir, Rinaldi. 2021. Pencocokan-String-2021.
<https://informatika.stei.itb.ac.id/~rinaldi.munir/Stmik/2020-2021/Pencocokan-string-2021.pdf>.diakses 26 April 2021 pukul 10.22 WIB
- [3] Youtube. https://www.youtube.com/watch?v=K8L6KVGG-7o&ab_channel=CoreySchafer.
Diakses 20 April 2021, pukul 21.30 WIB
- [4] Dharma Widya, 2019.PENERAPAN ALGORITMA PENCARIAN
KNUTH-MORRIS-PRATT (KMP) DALAM SISTEM INFORMASI PERPUSTAKAAN SMK
TI PRATAMA. Diakses pada 20 April 2021, Pukul 21.45 WIB